EXHIBIT 2

FILED UNDER SEAL

CLEMENT SETH ROBERTS (STATE BAR NO. 209203)
croberts@orrick.com
BAS DE BLANK (STATE BAR NO. 191487)
basdeblank@orrick.com
ALYSSA CARIDIS (STATE BAR NO. 260103)
acaridis@orrick.com
EVAN D. BREWER (STATE BAR NO. 304411)
ebrewer@orrick.com
ORRICK, HERRINGTON & SUTCLIFFE LLP
The Orrick Building
405 Howard Street
San Francisco, CA 94105-2669
Telephone: +1 415 773 5700
Facsimile: +1 415 773 5759

SEAN M. SULLIVAN (*pro hac* vice)
sullivan@ls3ip.com
MICHAEL P. BOYEA (*pro hac* vice)
boyea@ls3ip.com
COLE B. RICHTER (*pro hac* vice)
richter@ls3ip.com
LEE SULLIVAN SHEA & SMITH LLP
656 W Randolph St., Floor 5W
Chicago, IL 60661
Telephone: +1 312 754 0002
Facsimile: +1 312 754 0003

Attorneys for Sonos, Inc.

UNITED STATES DISTRICT COURT

NORTHERN DISTRICT OF CALIFORNIA

SAN FRANCISCO DIVISION

GOOGLE LLC,

Plaintiff and Counter-defendant,

v.

SONOS, INC.,

Defendant and Counter-claimant.

Case No. 3:20-cv-06754-WHA
Related to Case No. 3:21-cv-07559-WHA

REBUTTAL EXPERT REPORT OF DOUGLAS C. SCHMIDT

be responsible for playing back media from such "playback queue" at one time or another.

97. For example, the '033 Patent describes an embodiment in which a user listens to music on the user's MacBook Pro from an online media service, such as "turntable.fm or other virtual room that a user can enter to choose from a plurality of ***online disc jockeys (DJs) deciding what to play next*** …." The user then decides to play that music on the user's "household playback system" (comprising one or more "playback devices") by selecting "[a] button or other indicator … added to the turntable.fm Web application" that "switch[es] the content being played to the playback system for output (e.g., to the Sonos™ system rather than … the Mac Book™)." *Id.*, 12:65-13:11. In this example, the "playback queue" is "remote" of both the "control device" (MacBook Pro) and the "household playback system."

98. As another example, the '033 Patent describes an embodiment in which "a shared queue is ***provided between*** the local playback system and the third party application to keep the local system and application synchronized." *Id.*, 16:64-67. Again, in this example, the "playback queue" is "remote" of both the "control device" (computing device running the third-party app) and the "local playback system" (comprising one or more "playback devices").

99. As yet another example, the '033 Patent describes embodiments that "facilitate control of a local playback system ***from outside*** a household or other location at which the local playback network is configured" such that "a user can ***queue up music while away from*** his or her ***house***." *Id.*, 17:8-11. In this example, the "playback queue" is "remote" of at least the household "playback network" (comprising one or more "playback devices").

100. Sonos's Technology Tutorial provides an overview of how the example system architecture shown in Figure 7 enables a user to transfer playback from their smartphone to a home "playback device." I understand that Sonos sometimes refers to the technology of the '033 Patent as "Direct Control" or "Play to Sonos" technology.

VII. CLAIM CONSTRUCTION

A. "Playback Queue"

1. The Court's Summary Judgment Order

101. As discussed in my Opening Report, I understand that the Court construed the term

"playback queue" as "a list of multimedia content selected for playback" in the context of claim 13 of the '615 Patent. *See* 20-cv-6754, Dkt. 316 [Order Granting Motion for Partial Summary Judgment as to '615 Patent].

102. In reaching its construction, I understand that the Court reasoned (i) "a list of one is still a list," (ii) "nothing requires a 'playback queue' to contain plural multimedia items" and thus, "the list must contain at least one item, but not necessarily more than one," and (iii) a user need not "directly populate and manage the queue" and thus, "the list does not necessarily… require users to select content directly." *Id.*, 7-8. While the Court stated "[t]he patent repeatedly ***associates*** a queue with a 'list' or 'playlist,'" I take the Court's statement (especially the use of the word "associates") in the context of the Court's overall order to ***not*** mean that a "playlist" is ***equivalent*** to a "playback queue." *See also, e.g.*, *id.*, 7 ("[T]he specification repeatedly describes embodiments where a queue only contains a single audio track.").

103. In applying its construction for purposes of infringement, however, the Court further defined what is required to be a "playback queue." For instance, the Court started its infringement analysis by considering which queue was used by Google's accused apps running on the accused playback devices for playback – the "local" queue or the "cloud"/"remote" queue. *Id.*, 8-9. Put another way, the Court determined that the "playback queue" needed to be the queue that was used for playback of the list of multimedia content. In finding no infringement of claim 13 of the '615 Patent, the Court agreed with Google that the accused apps running on the accused playback devices "do not use a 'local playback queue.'" *Id.*, 8-10.

104. In its Order, the Court also indicated that the "playback queue" must have a complete list of all the multimedia items that are to be played back – a "subset" or "short list" of such items is not enough to be a "playback queue." *Id.*, 9 ("The groups of three items stored by the respective apps are not lists of multimedia content selected for playback."), 10 ("The passage thus distinguishes a local playback queue from the 'short list of tracks.'").

105. The Court also indicated in its Order that a queue was not a "playback queue" if it "merely provide[d] the means to *process* the list[] for playback." *Id.*, 10 (emphasis in original).

106. And finally, the Court concluded that, "[in] short," the "playback queue" is the

"queue [that] runs the show." *Id*.

107. Thus, according to the Court's Order, I understand that the claim term "playback queue" refers to a "list of multimedia content selected for playback" with the following characteristics:

- The playback queue is the list of media items that is used for playback;
- The playback queue contains the entire list of media items selected for playback;
- The playback queue is not being used merely to process the list of media items for playback; and
- The playback queue is the queue that "runs the show."

108. Based on the Court's Order and its requirements for a "playback queue," it appears that, in a system like the YTR System and Tungsten/Q System, there can only be ***one*** playback queue, which is either remote or local (but not both). I understand that Google has taken the same position. *See, e.g.*, Dkt. 343.02 [Sonos Inc.'s Opposition to Google LLC's Motion for Leave to Amend Invalidity Contentions Pursuant to Patent L.R. 3-6], 4-6; *infra* §XI.A.

2. "Multimedia Content" versus "Media Item"

109. In its Order, the Court rejected "Google's proposal to include the term 'multimedia item' in the construction" and instead construed "playback queue" in claim 13 of the '615 Patent to recite "multimedia content" because "[t]he claim uses the term 'multimedia content,' and there is no need to introduce additional ambiguity by importing a new term." Dkt. 316, 8.

110. I understand that Dr. Bhattacharjee has applied the Court's construction of the term "playback queue" provided in the context of claim 13 of the '615 Patent ***verbatim*** to the term "playback queue" provided in the context of the '033 Patent's claims.

111. As I noted in my Opening Report, however, the '033 Patent's claims do not recite the term "multimedia content" like the '615 Patent's claims do. Instead, the '033 Patent's claims recite the term "media item." For purposes of the '033 Patent, therefore, I will interpret the Court's construction of "playback queue" (provided in the context of claim 13 of the '615 Patent) as "***a list of one or more media items selected for playback***."

112. However, my opinions would remain the same under the Court's exact construction of "playback queue" provided in the context of claim 13 of the '615 Patent and applied verbatim by Dr. Bhattacharjee because a POSITA would understand that the term "multimedia content" is

A. Google & Dr. Bhattacharjee's Diametrically Inconsistent Positions Regarding Google's Systems' "Queue" Implementations

299. In connection with the Court's "patent showdown" procedure, I understand that Google filed a motion for summary judgment of non-infringement and invalidity of claim 13 of the '615 Patent. As noted above, I provided a declaration in support of Sonos's opposition to Google's summary judgment motion. I understand that the Court ultimately sided with Google on both issues of non-infringement and invalidity of claim 13 of the '615 Patent.

300. That said, in convincing the Court to rule in its favor, I understand that Google and Dr. Bhattacharjee took various positions that are in direct conflict with positions that Dr. Bhattacharjee is taking now in his attempt to invalidate the Asserted Claims of the '033 Patent. It is therefore my opinion that Google and Dr. Bhattacharjee's own words contradict Dr. Bhattacharjee's current opinion that "[t]he Court's order supports [his] opinion that the asserted claims of the '033 patent are invalid because the differences in claims language between the asserted claims of the '033 patent and Claim 13 of the '615 patent relate to obvious variations of the invention in Claim 13 of the '615 patent." Bhatta. Op. Report, ¶263.

1. Dr. Bhattacharjee's Opinions Contradict Google & Dr. Bhattacharjee's Representations to the Court

301. As an initial matter, I understand that the '615 Patent summary judgment dealt with Sonos's assertions that Google's implementations of its YouTube apps and Google Play Music (GPM) app as of May 2018 infringed claim 13 of the '615 Patent that recites "a local playback queue on [a] particular playback device." Critically, I understand that the primary references that Dr. Bhattacharjee presently relies on to attempt to invalidate the Asserted Claims of the '033 Patent are Google's predecessors of the YouTube and GPM systems: (i) the YTR System is the predecessor of the accused YouTube system and (ii) the Tungsten/Nexus Q system is the predecessor of the accused GPM system.[30] I understand that, in convincing the Court to rule in its favor on both non-infringement and invalidity of claim 13 of the '615 Patent, Google and Dr. Bhattacharjee made numerous, broad-sweeping characterizations regarding Google's predecessor systems relative to the accused YouTube and GPM systems that directly contradict Dr.

[30] *See, e.g.*, Bhatta. Op. Report, ¶¶217, 219, 233, 458.

Bhattacharjee's opinions here.

302. **First**, Google and Dr. Bhattacharjee repeatedly represented to the Court that Google's prior art systems ***used only*** a "local playback queue" on a playback device, while the accused YouTube and GPM systems ***use only*** a "remote playback queue" (also referred to by Google and Dr. Bhattacharjee as a "cloud queue"):

- "***[B]y 2013*** Google ***began to*** transition its applications to a 'cloud queue.' ***Prior to this transition***, however, ***Google's prior art products used a conventional 'local playback queue'*** architecture for managing music playback." 20-cv-6754, Dkt. 211 [Google's Motion for Summary Judgment Pursuant to the Court's Patent Showdown Procedure], 1;

- "[T]he YouTube Remote prior art product is a direct ancestor of the YouTube product Sonos accuses of infringement …. The ***key difference*** is that where the accused YouTube applications ***use … a cloud queue***, the prior art YouTube Remote ***used*** … ***a local queue***. Thus, Google's ***current (remote queue)*** products cannot infringe the patent, but its ***prior art (local queue)*** products invalidate that same patent." *Id.*, 2;

- "Google's accused systems … have long since moved away from a 'local playback queue' in favor of a playback queue stored in the cloud (*i.e.*, a 'cloud queue')." *Id.*, 3-4;

- "**[O]nly** older versions of [the MDx] protocol **stored the playback queue on the playback device**. When Google transitioned to Mdx Version 3, it **eliminated** the **playback queue on the playback device** in favor of a **cloud queue** in the [sic] maintaining it on the MDx server. *See* Ex. 2 at GOOG-SONOSWDTX-00041748 ('the queue is now maintained on ***the MDx server and not the TV***.'); Ex. 3 at GOOG-SONOSWDTX-00039988 ("MDx is ***the first*** server-backed ***Cloud queue*** at YT")." *Id.*, 5 (bold and italics emphasis added; bold and underlined emphasis added);

    - "***Version 3*** of the MDx protocol was ***created*** by January ***2014***. Ex. 5 (MDx Communication Protocol v3) at GOOG-SONOSWDTX-00037243. Thus, the decision to move to a Cloud Queue in MDx came just months after Google informed Sonos that it was moving to a Cloud Queue in GPM." *Id.*, 5 fn. 2.

- "But whereas the accused applications use MDx Version 3, the YTR prior art used the older MDx Version 1. ***This distinction is significant*** because … one of the changes that Google made in MDx Version 3 was to ***eliminate*** the ***playback queue on the playback device*** in favor of maintaining it in a ***Cloud Queue***…. Ex. 1 (Bhattacharjee Decl.), ¶¶49, 64 (***showing YTR prior art stores queue on playback device***)." *Id.*, 17 (some citations omitted);

- "**MR. VERHOEVEN [FOR GOOGLE]:** So if you have a phone – I have a phone, Your Honor. I create this playlist, this beautiful playlist. Okay? There's two ways you can deal with storing this playlist. You can store it on the speaker or you could store it in the Cloud. **THE COURT:** Why not store it on your own phone? **MR. VERHOEVEN:** Or you can store it locally on your phone, whith it doesn't – isn't even alleged to do…. ***It used to be local***, Your Honor. So the choice used to be, we've got to populate it here…." Transcript of Court Proceedings [Patent Showdown Summary Judgment Hearing] (July 13, 2022) ("Patent Showdown Hr'g Tr."), at 49-50.

- "**MR. VERHOEVEN [FOR GOOGLE]:** Yeah. And that just shows, [in the accused products] the phone is not involved in the processing of the queue or maintenance of the queue. The queue is maintained up here, the phone gives the instruction, and the receiver calls for the first item in the queue. But the queue is up here in the Cloud. ***It used to be down here in the [playback] device***, and for a variety of technical reasons, it moved to the Cloud, just like so many other things have moved to the Cloud. And ***so the queue used to be maintained in the remote device or the speaker playback device***. Sonos did it that way. ***Google did it that way***." *Id.*, at 63-64.

- "Whereas early versions of the MDx protocol—***such as that used in the YouTube Remote prior art*** *…*—***maintained a playback queue on the Screen***, in Version 3 Google changed MDx to eliminate the playback queue on the Screen in favor of maintaining it on the MDx server[.]" Declaration of Samrat Bhattacharjee, PH.D (dated April 14, 2022) ("Bhatta. Showdown Decl."), ¶49;

- "In the accused YouTube system, the playback queue is not stored on a playback device, such as the playback device. Rather, as already mentioned, it is stored in the Cloud Queue. *See, e.g.*, supra ¶49. Indeed, the protocol used by the accused YouTube system for controlling playback is Version 3 of the MDx protocol, which was ***changed from earlier version of MDx***—such as ***Version 1 that is used in the prior art YT Remote*** discussed in Section VII—to ***eliminate the "local playback queue" in favor of a Cloud Queue***." *Id.*, ¶64;

- "That the accused ***YouTube system does not use a 'local playback queue'*** is further evidenced by the fact that in the accused system if the MDx servers were to go offline, playback of the playlist would not be possible. In contrast, ***in a system that stored the 'playback queue' locally on the device, the playback device could continue to play the queue***—whether a remote copy was available or not. As an example, ***consider the prior art YouTube Remote System*** that I discuss in this Declaration (Section VII): ***the playback device in the YouTube Remote prior art stores a list of videoIDs for the playlist*** and is thus ***capable of playing back the playlist even if the MDx server were not available***." *Id.*, ¶74;

- *Id.*, ¶¶139-45;

- "In fact, as it relates to this limitation, the only material difference between the prior art YTR system and the YouTube application is that ***the prior art YTR system*** actually ***stores a 'local playback queue*** on the particular playback device' (an array that stores the ordered list of videoIDs in the playlist), while the accused YouTube system does not (Section VI)." *Id.*, ¶180;

- "The list of videoIDs in the setPlaylist message are ***stored locally*** on the Screen as an array (a 'local playback queue on the particular playback device')." *Id.*, ¶181.

303. **Second**, Google and Dr. Bhattacharjee repeatedly represented to the Court that the existence of a "local playback queue" in a system is ***mutually exclusive*** of a "remote playback queue" (or "cloud queue") and *vice versa*:

- "The playlist where the accused products store the ordered list of items to be played is in the cloud queue … rather than any 'local' playback queue." 20-cv-6754, Dkt. 211 [Google's Motion for Summary Judgment Pursuant to the Court's Patent Showdown Procedure], 8;

- "The [accused data construct] is merely a window of three locally cached items from the 'playback queue'—***it is not the 'playback queue' itself***. The 'playback queue'… remains in the Cloud." *Id.*, 15;

- "These GPM playback queues are stored in the cloud queue; they cannot exist in the [accused data construct] because it is merely a windowed view of the previous, current and next media items in the playback queue which is stored in the cloud." *Id.*, 15;

- "At best, Sonos has shown that the [playback device] stores individual items from the playback queue —the previous, current, and next videoID—rather than the queue itself." 20-cv-6754, Dkt. 278 [Google's Reply in Support of Google's Motion for Summary Judgment Pursuant to the Court's Patent Showdown Procedure], 2.

- "**THE COURT:** Now, as I read the paperwork, what you do have is the last played, the current one, and the next one? **MR. VERHOEVEN [FOR GOOGLE]:** Right. **THE COURT:** Why isn't that enough to satisfy the queue? **MR. VERHOEVEN:** Because that's not a queue. **THE COURT:** Why isn't it? **MR. VERHOEVEN:** That's the processing of the queue that's resident on the Cloud." Patent Showdown Hr'g Tr., 42-43;

- "**THE COURT:** All right. So, Mr. Verhoeven, the argument against you here is that -- there are two queues: There is one in the Cloud, and there is another one on the speaker. The local playback queue is on the speaker, and all it needs to know is what is the next one. And so that's good enough for its purposes and just calls the next one. What do you say to that? **MR. VERHOEVEN [FOR GOOGLE]:** I say that that's unpersuasive in the extreme . . .We're talking about ***a*** queue. ***Somebody has to own that queue***. Somebody has to ***maintain that queue***. Somebody has to be in charge of ***that queue***. Who is it? It's the Cloud. That's why it's called ***the*** Cloud ***queue***. ***The queue*** is maintained in the Cloud. If you want to know -- if you're the speaker and you want to know the next item in ***the queue***, you ask the Cloud because the Cloud maintains ***the queue***." *Id.*, 59-60;

- "In the accused YouTube system, ***the playback queue*** is not stored on a playback device, such as the playback device. Rather, as already mentioned, it is stored in the Cloud Queue. *See, e.g.*, supra ¶49. Indeed, the protocol used by the accused YouTube system for controlling playback is Version 3 of the MDx protocol, which was ***changed from earlier version of MDx***—such as ***Version 1 that is used in the prior art YT Remote*** discussed in Section VII—to ***eliminate the "local playback queue" in favor of a Cloud Queue***." Bhatta. Showdown Decl., ¶64;

- "The GPM system does not infringe because it uses a Cloud Queue, not a 'local playback queue on the particular playback device.'" *Id*., ¶113; *see also*, *id.*, ¶¶115-20;

304. I understand that the Court sided with Google based on Google and Dr. Bhattacharjee's representations and ordered summary judgment that the YTR System invalidated claim 13 of the '615 Patent but that the accused YouTube apps did not infringe claim 13 of the '615 Patent. 20-cv-6754, Dkt. 316 [Order Granting Motion for Partial Summary Judgment as to '615 Patent], 10. In this regard, I understand that the Court accepted Google and Dr. Bhattacharjee's representations that a system cannot have both a "local playback queue" and a "remote playback queue"/"cloud queue" because "locally-stored information is merely a mirror reflecting a subset of what is happening in the cloud queue." *Id.*, 9-10. I understand that, according to the Court, groups of three items (such as that stored by a playback device when used with the accused YouTube apps) did not constitute a "playback queue" because they "merely provide the means to *process* the lists for playback. In short, the cloud queue runs the show." *Id.*, 10.

305. In view of Google and Dr. Bhattacharjee's representations to the Court, I disagree with Dr. Bhattacharjee opinion that "[t]he Court's order supports [his] opinion that the asserted claims of the '033 patent are invalid …." Bhatta. Op. Report, ¶263.

2. Dr. Bhattacharjee's Opinions Contradict His Prior Opinions Regarding the Same Predecessor Systems

306. In advance of the previously-scheduled "patent showdown" trial, Dr. Bhattacharjee opined in his Opening and Rebuttal Showdown Reports that both the YouTube Remote and Tungsten/Nexus Q systems involved a "local playback queue." In view of such opinions and Google and Dr. Bhattacharjee's ***representations to the Court*** that the existence of a "local playback queue" in a system is mutually exclusive of a "remote playback queue," Dr. Bhattacharjee's opinions in his current report that both the YouTube Remote and Tungsten/Nexus Q systems involved a "remote playback queue" are simply not credible.

307. Specifically, as shown in the example passages below, Dr. Bhattacharjee repeatedly and unequivocally opined in his Opening and Rebuttal Showdown Reports that the YTR System involved a "local playback queue" to the exclusion of a "remote playback queue":

- "Indeed, the accused applications are similar to the ***YouTube Remote*** and Tungsten/NexusQ prior art, with the exception that the prior art ***stored the playback queue locally on playback devic***e (as required by claim 13) while the accused applications moved the playback queue to the cloud and thus do not infringe." Bhatta. Rebuttal Showdown Report, ¶387;

- "[A]s I discussed in my opening report, Google ***stored the playback queue locally on its receiver devices in its prior art products***. ... However, in 2013 Google worked with Sonos to move the playback queue to the cloud." *Id*., ¶297;

- "Whereas ***Version 1 and 2*** of the MDx protocol maintained ***a playback queue on the Screen***, in Version 3 Google changed MDx to eliminate the playback queue on the Screen in favor of maintaining it on the MDx server: "the queue is now maintained on the MDx server and not the TV." *Id*., ¶81;

- "[A]s I showed in my opening report Version 1 of the MDx protocol differs from Version 3 of the MDx protocol in that in ***Version 1*** of the MDx protocol a YouTube receiver received a setPlaylist message containing the list of videoIds for the items in ***a playback queue***, whereas in ***Version 3*** of the MDx protocol the YouTube receiver receives the videoId for the item that should begin playing and the list of the rest of the videoIds is stored on the MDx server in ***a cloud queue***." *Id*., ¶78;

- "Like Version 1 of the MDx protocol, in ***Version 2*** of the MDx protocol a playback device received a setPlaylist message containing the list of videoIds in ***the playback queue***." *Id*., ¶80;

- "In ***Version 1*** (and ***Version 2*** of the MDx protocol) this involved the alleged playback device receiving a setPlaylist message that included the list of videoIds for the videos in ***the playback queue that the playback device stored in an array on the receiver device***. ... But in ***Version 3*** of the MDx protocol, which is used by the accused devices, Google changed the MDx protocol and the setPlaylist message. Rather than send a setPlaylist message to the receiver that contains a list of videoIds for the videos in the playlist, YouTube now stores that list of videoIds in ***a cloud queue*** and the receiver device can fetch the identity of the next song from the cloud queue one-by-one." *Id*., ¶¶173-174;

- "Indeed, documents describing the accused YouTube System and its MDx protocol confirm that the playback queue is not stored on the receiver device ***beginning*** in Version 3 of the MDx protocol." *Id*., ¶175;

- "In particular, when the user taps on the 'Connect' button to transfer playback, the ***YTR application sends*** a message to the Lounge Server with ***the current playlist***. GOOG-SONOSWDTX-00041837 ('remote sends the current playlist to the screen.')." Bhatta. Op. Showdown Report, ¶188; *see also id.*, ¶¶190-91;

- "The Leanback Screen receives the setPlaylist message and commences to play the video. In particular, the 'setPlaylist' message invokes the remoteSetPlaylist function (LeanbackModule.as, line 475). This function ***creates a new local playlist*** (a 'station') from the list of videoIds that are received (line 477). ***The playlist is stored locally*** as an array of variable size. (LeanbackModule.as, line 510-525, remotefeed.as, line 8, staticfeed.as, line 12, feed.as, line 22)." *Id.*, ¶192; *see also id.*, ¶¶193-99;

- "[T]he only material difference between the prior art YTR system and the YouTube application is that ***the prior art YTR system actually stores a 'local playback queue*** on the particular playback device' (an array that stores the ordered list of videoIDs in the playlist) under Google's proposed constructions (*see* Section [sic]), while the accused YouTube system does not. YT Remote API () at 3 ('setPlaylist' includes 'set of video ids of any length, separated by commas')." *Id.*, ¶231;

- "Indeed, in the prior art YTR system upon transferring playback the YTR application caused a Lounge Server ('the one or more first cloud servers') to send a setPlaylist message to the paired Leanback Screen(s) ('the particular playback device') with a list of videoIDs for the playlist ('multimedia content' and 'resource locators' according to Sonos). ***The list of videoIDs*** in the setPlaylist message ***are stored locally on the Screen*** as an array (***a 'local playback queue*** on the particular playback device')." *Id.*, ¶232.

308. In this way, Dr. Bhattacharjee previously opined that the YTR System involved a "local playback queue," while the Accused YouTube system does not (because it involves a "remote playback queue") and that this was a "material difference" between the systems. *See, e.g.*, Bhatta. Op. Showdown Report, ¶231. In other words, Dr. Bhattacharjee previously opined that the

YTR System did not disclose a "remote playback queue," and the existence of a "local playback queue" in a system is mutually exclusive of a "remote playback queue."

309. Moreover, Dr. Bhattacharjee previously opined in his Rebuttal Showdown Report that "[w]hether or not some hypothetical system may store the playback queue in a cloud queue and also store a copy of that playback queue locally on the receiver device is irrelevant because ***YouTube does not store the playback queue at both locations***." Bhatta. Rebuttal Showdown Report, ¶114.

310. Given the foregoing, it is my opinion that Dr. Bhattacharjee's opinions in his current report that the YTR System discloses a "remote playback queue" are not credible. *E.g.*, *compare* Bhatta. Op. Report, ¶328 ("After receiving the SET_PLAYLIST and SET_VIDEO messages from the Lounge server in either of the above scenarios, the Screens are configured to communicate with the cloud-based computing system in order to obtain data identifying a next one or more media items ***that are in the remote playback queue.***"), *with* Bhatta. Op. Showdown Report, ¶189 ("The list of videoIDs in the setPlaylist message are ***stored locally on the Screen*** as an array (***a*** '***local playback queue*** on the particular playback device')." *Id.*, ¶232; *see also, e.g.*, Bhatta. Op. Report, ¶¶169, 171, 173, 181, 186, 298, 300.

311. Likewise, as shown in the example passages below, Dr. Bhattacharjee repeatedly opined in his Opening and Rebuttal Showdown Reports that the Tungsten/Nexus Q system involved a "local playback queue" to the exclusion of a "remote playback queue":

- "Indeed, the accused applications are similar to the YouTube Remote and ***Tungsten/NexusQ*** prior art, with the exception that the prior art ***stored the playback queue locally on playback devic***e (as required by claim 13) while the accused applications moved the playback queue to the cloud and thus do not infringe." Bhatta. Rebuttal Showdown Report, ¶387;

- "[A]s I discussed in my opening report, Google ***stored the playback queue locally on its receiver devices in its prior art products***. ... However, in 2013 Google worked with Sonos to move the playback queue to the cloud." *Id*., ¶297;

- "As I showed in my opening report, at the time of the alleged invention in 2011, the Project Tungsten/NexusQ prior art (and the Music2 application that was part of this system) used a ***'local playback queue.'*** In other words, the Tungsten/NexusQ prior art allowed users to add tracks to a playback queue. The playback queue was then stored locally on the playback device. Storing the playback queue locally on the playback device, as is claimed in the '615 patent and was done by the Tungsten/NexusQ prior art, is ***in sharp contrast*** to the accused applications which use ***a cloud queue***. Users of the accused applications may create a playback queue, which can then be stored on an Internet server (also referred to as a 'cloud'server) rather than the playback device. The playback device can then playback the cloud queue by caching information about the next item in the queue rather than storing the entire playback queue." *Id*., ¶114;

- "All blocks [i.e., Tungsten devices] have both receiver 'Rx' and transmitter 'Tx' functionalities, meaning each Tungsten can be a Tx and an Rx… ***The Tx maintains an ordered list of multimedia items*** that is selected by the user ***for playback*** (discussed in greater detail below) ***and is thus a 'local playback queue'*** under both Google and Sonos's interpretation of the term. A user can select and group different (1 or more) Rx blocks to be associated with the Tx block - the Tx block itself is also an Rx. ***The Tx that holds the local playback queue*** can also function as an Rx and play back media. For example, when a user selects a single Tungsten (block) for playback, the single Tungsten functions as both a Tx and an Rx, meaning ***the Tungsten stores the local playback queue*** and ***plays back this local playback queue***." Bhatta. Op. Showdown Report, ¶270;

- "As just mentioned, ***Tungsten playback devices maintain a 'local playback queue.'*** *See* athome/google_athome/blockhead/src/com/timoco/blocks/blockhead/db/PlaylistProvider.java. The playlist is stored in a database named 'playlist.db' (line 183), and contains a table called 'playlist', with columns for song-id, artist, album, title, etc. (lines 167-177). ***The PlaylistProvider.java file contains methods*** for inserting, updating, and querying this database, which is ***consistent with the use of a local playback queue.*** In particular, the tracks are stored in order within the database, and can be used to find tracks in order, e.g., to ***play a track*** at a specific index ***within the local playback queue***. *See* the doPlayAtIndex function in athome/google_athome/blockhead/src/com/timoco/blocks/blockhead/services/TXService.java (lines 318-331)." *Id.*, ¶271; *see also id.,* ¶375 ("The Tungsten source code as it existed on July 14, 2011 confirms this. ***Tungsten playback devices maintain a local playback queue***."); head/db/PlaylistProvider.java. The playlist is stored in a database named 'playlist.db' (line 183), and contains a table called 'playlist', with columns for song-id, artist, album, title, etc. (lines 167-177). blocks/blockhead/services/TXService.java (lines 318-331).

- "A Tx block ***locally stores a playback queue*** of song IDs…. As I explained above, because each Tungsten can be both a Tx and an Rx, when a user has only one Tungsten or selects only one Tungsten, ***that Tungsten both stores the local playback queue, and also plays back the local playback queue***." *Id.*, ¶278;

- "Indeed, the December 29, 2011 capture of the Tungsten source code that I reviewed discloses that the Tungsten system at the time included the Music 2 application on both a control device, e.g., a user's phone, and on the Nexus Q device… A user could play music on the control device using the Music 2 application, then select a Nexus Q for playback. Once the Nexus Q started playing, playback stopped on the control device. The ***Tungsten devices*** also ran the Music2 app and ***maintained a local play queue***. *See* AtHomeMusicServer.java, line 81. The ***queue was maintained*** as a linked list (AtHomePlayQueue.java, line 42), and contained information such as song id, album, etc." *Id.*, ¶289;

- "[T]he December 2011 Tungsten system included one or more first cloud servers that added multimedia content to ***a local playback queue on a particular playback device (a Tungsten device)***, wherein adding the multimedia content to the local playback queue comprised the one or more first cloud servers (e.g., Skyjam FE servers) adding, to the local playback queue, one or more resource locators (e.g., streaming URLs) corresponding to respective locations of the multimedia content at one or more second cloud servers of a streaming content service (e.g., Bandaid servers)." *Id.*, ¶360;

- "The December 2011 Tungsten source code confirms that Tungsten devices cause 'one or more first cloud servers to add multimedia content to ***a local playback queue*** on the particular playback device, wherein adding the multimedia content to the local playback queue comprises the one or more first cloud servers adding, to the local playback queue, one or more resource locators corresponding to respective locations of the multimedia content at one or more second cloud servers of a streaming content service.'" *Id.*, ¶364;

- "The Tungsten devices run the Music2 app, which I understand eventually became Google Play Music. ***Tungsten devices maintain a local playback queue***. *See e.g.,* AtHomeMusicServer.java, line 81. The queue was maintained as a linked list (AtHomePlayQueue.java, line 42), and contained, for the ordered list of items in the queue, information such as song id, album, etc." *Id.*, ¶365.

312. In his current report, Dr. Bhattacharjee opines that the Tungsten/Nexus Q system allegedly discloses ***both*** a "local playback queue" ***and*** a "remote playback queue." *See, e.g.,* Bhatta. Op. Report, ¶786 ("The version of the GPM application (also referred to as the Music2 application) that is prior art allowed users to play back music on Google Tungsten/NexusQ devices, including ***cloud-hosted playlists*** (such as an album playlist or MagicPlaylist). ***A copy of the playlist was also stored on the playback device (the Tungsten/NexusQ device)***."). However, as discussed, I understand that Google and Dr. Bhattacharjee repeatedly and vehemently represented to the Court that (i) Google's prior art systems did not use a "remote playback queue" and instead, "used a conventional 'local playback queue' architecture" (Dkt. 211, 1) and (ii) the existence of a "local playback queue' in a system is mutually exclusive of a "remote playback queue." In view of such representations to the Court, I disagree with Dr. Bhattacharjee's opinion that the Tungsten/Nexus

Q system allegedly discloses both a "local playback queue" and a "remote playback queue" (*id.*) and that "[t]he Court's order supports [his] opinion that the asserted claims of the '033 patent are invalid …." *Id.*, ¶263.

3. Dr. Bhattacharjee's "Obvious Variations" Opinions Are Not Credible

313. I further disagree with Dr. Bhattacharjee's opinion that the claims of the '033 Patent and the '615 Patent are "obvious variations" of one another for various reasons. For instance, Dr. Bhattacharjee's opinion that "the asserted claims of the '033 patent are invalid because the differences in claims language between the asserted claims of the '033 patent and Claim 13 of the '615 patent relate to obvious variations of the invention in Claim 13 of the '615 patent" (Bhatta. Op. Report, ¶263) is unsupported, conclusory, and directly contradicted by Google and Dr. Bhattacharjee's prior representations to the Court. *See, e.g.*, 20-cv-6754, Dkt. 211 [Google's Motion for Summary Judgment Pursuant to the Court's Patent Showdown Procedure], 11 ("Google's approach— ***using a cloud queue***—is ***fundamentally different*** than the "***local playback queue***" system that Sonos claimed. *See* Ex. 1 (Bhattacharjee Decl.), ¶¶86-88…. Therefore a local and remote playback queue are '***very different***' ways of using queues in the multimedia context and are not equivalents. *Id.*, ¶¶86-89."); Bhatta. Showdown Decl., ¶¶86-89.

314. Dr. Bhattacharjee also previously opined in his Rebuttal Showdown Report that a POSITA would understand that there is a "***substantial difference***" between a "cloud queue" and a "local playback queue." Specifically, Dr. Bhattacharjee opined:

> A person of skill in the art would understand that there is ***substantial difference*** between a cloud queue (which involves fetching the identity of the next itemin the cloud queue one-by-one) and a local playback queue. For example, a cloud queue provides centralized storage of the playback queue. As a result, it can support playback and interactions from a large number of users and can be shared with many devices. Moreover, because the centralized storage is in the cloud, the cloud queue is not restrained by the capabilities of the playback device's hardware (e.g., memory), and the queue is not lost when a playback device fails. Thus, a cloud queue can allow users to maintain a far greater number of items in a playback queue. A cloud queue also permits multiple sender devices to edit and manage the queue, even where they are not connected to the playback device through a local area network.

Bhatta. Rebuttal Showdown Report, ¶267; *see also id.*, ¶271 ("there is a ***substantial difference*** between using a local playback queue and a cloud queue."), ¶272 (same). In fact, Dr. Bhattacharjee

criticized my prior infringement opinions regarding claim 1 of the '615 Patent because I allegedly "fail[ed] to appreciate and give meaning to the difference between a 'remote' and a 'local' playback queue." *Id*., ¶208.

315. In view of Google and Dr. Bhattacharjee's prior representations, I simply do not find Dr. Bhattacharjee's present opinions to be credible.

316. Moreover, Dr. Bhattacharjee's assertion in his report that "a [POSITA] would understand that the disclosure regarding a local playback queue ***cannot*** disclose a remote playback queue" (Bhatta. Op. Report, ¶689) directly contradicts his opinion that the claims of the '033 Patent and the '615 Patent are "obvious variations" of one another. *See also id.*, ¶264 (asserting "[t]he primary difference is that Claim 12 of the '033 patent recites a 'remote playback queue' rather than the 'local playback queue' recited in Claim 13 of the '615 patent."), ¶816 (asserting the "primary difference" is that "[t]he '033 patent receives a 'remote playback queue,' while the '615 patent recites a 'local playback queue'").

317. Further, contrary to Dr. Bhattacharjee's opinion that the claims of the '033 Patent and the '615 Patent are "obvious variations" of one another, a POSITA would have understood that Dr. Bhattacharjee's proposed modification to add a "remote playback queue" to a system that purportedly had a "local playback queue" would have resulted in altering the system's principles of operation (which would have discouraged a POSITA from performing the proposed modification), and/or would have required modifying how the system already solved a particular problem, despite the fact that a POSITA having common sense would not have sought to solve a problem that was already solved.

318. For at least the foregoing reasons, I disagree with Dr. Bhattacharjee's opinion that "[t]he Court's order supports [his] opinion that the asserted claims of the '033 patent are invalid because the differences in claims language between the asserted claims of the '033 patent and Claim 13 of the '615 patent relate to obvious variations of the invention in Claim 13 of the '615 patent." Bhatta. Op. Report, ¶263.

4. The Court's Construction Does Not Justify Dr. Bhattacharjee's Inconsistent Positions

319. I understand that Google has attempted to blame the Court's claim construction of

"playback queue" as the justification for Google and Dr. Bhattacharjee's diametrically inconsistent positions. *See, e.g.*, Dkt. 336 [Google LLC's Motion for Leave to Amend Invalidity Contentions Pursuant to Patent L.R. 3-6]; 355 [Google LLC's Reply in Support of Its Motion for Leave to Amend Invalidity Contentions Pursuant to Patent L.R. 3-6]. Specifically, in seeking to amend its invalidity contentions to change course on its positions with respect to YouTube Remote's implementation of a "playback queue," I understand Google asserted that the Court not limiting its construction to content "selected by the user" opened the door for Google to contend that YouTube Remote's purported "party mode" feature that purportedly involved plural users selecting content for playback involved a "remote playback queue":

> [T]he Court rejected Sonos's broad construction of "playback queue" that encompassed any "data construct" that holds information corresponding to media for playback (including a single variable). *Id.* at 5-8. The Court also rejected Google's narrower construction of "[a]n ordered list of multimedia items that is selected by the user for playback." *Id.* The Court instead construed the term "playback queue" to mean "[a] list of multimedia content selected for playback." *Id.* at 5. In doing so, ***the Court broadened the types of playlists that would fall within the scope of the term "playback queue."*** For instance, the Court held that the "playback queue" was not restricted to a playlist in which the content was "selected by the user for playback," and instead ***covered playlists selected by a group of users for playback*** (such as a party playlist).

Dkt. 336, p. 6. It is my opinion that Google's assertion is egregiously false.

320. I understand that the Court rejected Google's prior argument that a "playback queue" is limited to user selected content, as opposed to content selected by a system or service (or some other non-user). *See, e.g.*, Dkt. 316 [Order Granting Motion for Partial Summary Judgment as to '615 Patent], 7-8 ("Sonos asserts that the content in the queue need not be selected directly by a user. Google's position, by contrast, is that a user must directly populate and manage the queue. Google's argument does not persuade…. [T]he specification also repeatedly describes embodiments in which the third-party application (such as Google Play Music) dictates what media content is in the queue."). Google ***never*** argued that its prior construction only allowed for a ***single*** user to select a list of content but ***excluded multiple*** users from selecting the list. *See, e.g.*, Dkt. 200 [Google LLC's Responsive Claim Construction Brief], 16-17.

321. Consequently, it is my opinion that the Court's construction that declined to limit a "playback queue" to user selected content provides no justification for Google and Dr.

Bhattacharjee to now take diametrically inconsistent positions. This makes sense because whether a "playback device" in a system plays back from a "local playback queue" or a "remote playback queue" does not depend on whether the content stored in that "playback queue" was user-selected or otherwise.

322. Indeed, as discussed below, Dr. Bhattacharjee points to the same respective functionality in the YouTube Remote and Tungsten/Nexus Q systems for satisfying a "playback device" playing from a "remote playback queue" in connection with "party mode" and a "Magic Playlist" as Dr. Bhattacharjee pointed to for satisfying a "playback device" playing from a "local playback queue." Moreover, Dr. Bhattacharjee cannot dispute that the "party mode" in YouTube Remote involved user selection of content for playback and the "Magic Playlist" in Tungsten/Nexus Q was triggered off of a user's selection for playback. *See, e.g.*, https://www.youtube.com/watch?v=OxzucwjFEEs&t=2808s 28:07-28:45, 31:09-31:16 (showing user manually selecting "Make instant mix" option from particular song). Thus, Google and Dr. Bhattacharjee have no justification for their inconsistent positions.

B. YouTube Remote (YTR) System + YouTube Remote Patent

323. With respect to validity, I understand that Dr. Bhattacharjee has asserted opinions that the Asserted Claims of the '033 Patent are anticipated by the YouTube Remote ("YTR") System or, alternatively, obvious based on the YTR System in view of (i) the general knowledge of a POSITA, (ii) the '998 Patent, (iii) Apple AirPlay, (iv) U.S. Publication No. 2011/0131520 to Al-Shaykh ("Al-Shaykh"), (v) the Tungsten/NexusQ Systems, and/or (vi) U.S. Patent No. 8,724,600 to Ramsay ("Ramsay"). I disagree with Dr. Bhattacharjee's opinions.

324. It is my opinion that Dr. Bhattacharjee fails to establish that the Asserted Claims of the '033 Patent are anticipated by the YTR System. More specifically, it is my opinion that Dr. Bhattacharjee fails to establish that the YTR System anticipates the Asserted Claims of the '033 Patent because he fails to establish that it discloses at least claim limitations 1.4-1.9, 2.1, 4.1, 7.1-7.2, 11.1, 12.1-12.6, 13.1, and 16.1.

325. It is also my opinion that Dr. Bhattacharjee fails to establish that the Asserted Claims of the '033 Patent are obvious based on the YTR System in view of (i) the general knowledge of a

Dated: January 13, 2023

DOUGLAS C. SCHMIDT